

INICIAR SESIÓN

NUESTROS PLANES

TODOS LOS
CURSOS

FORMACIONES

CURSOS

PARA
EMPRESAS

ARTÍCULOS DE TECNOLOGÍA > DATA SCIENCE

Cómo eliminar filas y columnas con Pandas en Python



Yuri Matheus

20/10/2020

Cuando trabajamos con datos, especialmente en la **fase de exploración de datos**, realizamos muchos cambios en el **dataset**.

Hay columnas que no queremos, o que no necesitamos usar, o filas que no contienen datos válidos que deben eliminarse.

Pero, ¿cuánto más datos tenemos, no es mejor?

Cuando analizamos datos, principalmente para entrenar **modelos de inteligencia artificial**, algunos datos pueden perjudicar en lugar de ayudar. Por lo tanto, tenemos que usar solo los datos que tengan sentido para el modelo.

¡Excelente! Ya sabemos que necesitamos tratar los datos, pero ¿cómo podemos hacer eso, cómo eliminar los datos?

Eliminar columnas en pandas

Tengo un dataset sobre propinas que dan los usuarios a los taxistas. Vamos a utilizar [Pandas](#) para nuestro análisis, Pandas es una biblioteca que facilita la manipulación de datos. En Pandas, tenemos un tipo de dato llamado DataFrame. Podemos decir que un DataFrame es como una planilla de Excel. Es decir, tenemos filas y columnas.

Vamos a realizar algunos análisis en nuestro dataset de propinas, llamado `tips`. Podemos ver el comienzo del contenido de la planilla utilizando el método `head` del data frame:

```
tips.head()
```

	total_bill	tip	smoker	day	time	size
0	16.99	1.01	No	Sun	Dinner	2
1	10.34	1.66	No	Sun	Dinner	3
2	21.01	3.50	No	Sun	Dinner	3
3	23.68	3.31	No	Sun	Dinner	2
4	24.59	3.61	No	Sun	Dinner	4

Podemos ver que el **DataFrame** se parece mucho a una planilla o a una tabla. Cada columna también conocida como variable tiene un nombre, y cada fila, conocida como observación, tiene un índice y los datos que representan cada variable.

Para el análisis que quiero hacer, no necesito saber si la persona fuma o no, por lo que no es necesario analizar la columna de **smoker**, ¿cómo podemos eliminarla?

En Pandas, hay un **método drop** para DataFrame. En este método, podemos pasar el índice de la fila que queremos eliminar:

```
>>> # en el
>>> tips.drop(2)
```

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2
1	10.34	1.66	Male	No	Sun	Dinner	3
3	23.68	3.31	Male	No	Sun	Dinner	2
4	24.59	3.61	Female	No	Sun	Dinner	4
5	25.29	4.71	Male	No	Sun	Dinner	4
6	8.77	2.00	Male	No	Sun	Dinner	2
7	26.88	3.12	Male	No	Sun	Dinner	4
8	15.04	1.96	Male	No	Sun	Dinner	2
9	14.78	3.23	Male	No	Sun	Dinner	2
10	10.27	1.71	Male	No	Sun	Dinner	2
11	35.26	5.00	Female	No	Sun	Dinner	4

Con este comando, eliminamos la fila con índice 2. Pero en realidad no queremos eliminar una fila, sino una columna. Para hacer esto, simplemente pasamos el parámetro `columns`, este parámetro recibe una lista con el nombre de las columnas que queremos eliminar, `smoker` en nuestro caso.

Después de esto, mostramos nuevamente los datos:

```
tips.drop(columns = ['smoker'])
tips.head()
```

	total_bill	tip	smoker	day	time	size
0	16.99	1.01	No	Sun	Dinner	2
1	10.34	1.66	No	Sun	Dinner	3
2	21.01	3.50	No	Sun	Dinner	3
3	23.68	3.31	No	Sun	Dinner	2
4	24.59	3.61	No	Sun	Dinner	4

La columna sigue ahí. ¿Qué sucedió?

Por defecto, las **operaciones que realizamos en un DataFrame no se aplican directamente en ese conjunto de datos**. Es decir, cada vez que eliminamos una columna, fila, etc., lo que hace Pandas es devolver un nuevo DataFrame con los cambios aplicados. Es decir, el DataFrame original permanece intacto.

Lo que podemos hacer para resolver esto es asignar este nuevo DataFrame que el método `drop()` retorna a la misma variable `tips`:

```
tips = tips.drop(columns = ['smoker'])
tips.head()
```

	total_bill	tip	day	time	size
0	16.99	1.01	Sun	Dinner	2
1	10.34	1.66	Sun	Dinner	3
2	21.01	3.50	Sun	Dinner	3
3	23.68	3.31	Sun	Dinner	2
4	24.59	3.61	Sun	Dinner	4

¡Super! Logramos eliminar la columna que necesitábamos. Ahora, pasemos a otra parte del procesamiento de datos. En el análisis que estoy haciendo, quiero analizar las propinas de aquellos recorridos donde hubo un grupos de pasajeros. Ese dato lo tenemos en nuestra columna `size` de nuestro DataFrame, se considera un grupo cuando en esta columna el valor es 2 o más.

¿Cómo podemos hacer para eliminar las filas que no cumplen con esta condición?

Eliminar filas

Sabemos que podemos usar el método **`drop()`** para eliminar una fila. Pero, ¿cómo eliminar todas las filas que contienen un solo pasajero?

Podemos recorrer e iterar a través de cada fila en el DataFrame y verificar si el número de pasajeros es mayor que uno. Si es así, conservamos esos datos, de lo contrario, los eliminamos del DataFrame.

Pero Pandas ya tiene varias formas de trabajar con DataFrames. Lo que podemos hacer es crear un filtro en el que se devuelva un nuevo DataFrame con los datos ya filtrados.

De la misma forma Pandas nos permite filtrar un DataFrame por sus datos de columna con condiciones en las filas. Por ejemplo, para el dataframe propinas (**`tips`**) capturamos la columna que define el número de pasajeros (**`size`**), en base a eso, filtramos las líneas en las que la columna `size` es mayor que uno:

```
filtro = tips['size'] > 1
```

Ahora, simplemente pasamos este filtro como selector en el DataFrame:

```
filtro = tips['size'] > 1  
propina_en_grupo = tips[filtro]
```

Este filtro nos devuelve un DataFrame con todas las filas donde el número de pasajeros es mayor que 1. Es decir, los datos que pasan a través del filtro definido:

```
propinas_en_grupo.head()
```

	total_bill	tip	day	time	size
0	16.99	1.01	Sun	Dinner	2
1	10.34	1.66	Sun	Dinner	3
2	21.01	3.50	Sun	Dinner	3
3	23.68	3.31	Sun	Dinner	2
4	24.59	3.61	Sun	Dinner	4

Para saber más

Los filtros utilizados pueden ser más complejos. Podemos usar estructuras condicionales, como **AND** y **OR**, y verificar el valor de más de una columna. Además, podemos hacer otros tipos de filtros.

¿Cómo verificar si alguna fila tiene un valor nulo o un valor de un tipo diferente al esperado?. Para esto, se puede utilizar el concepto de **rich comparison**.

Además de la eliminación, podemos realizar otras operaciones con DataFrames, como ser: **operaciones que involucran strings, renombrar columnas y agregar nuevas columnas, unir un DataFrame a otro, entre varias otras operaciones.**

¿Qué tal aprender más sobre **Pandas** y sus diversos recursos? Entonces, ¡Mira nuestros cursos de [Python para Data Science](#) aquí en [Alura](#)!

Para seguir expandiendo tus conocimientos descarga nuestro ebook gratuito de Data Science a través del siguiente link: <https://lp.caelum.com.br/alura-latam-leads-ebook-data-science>

Puedes leer también:

- [Análisis de datos: analizando mi distribución con tres alternativas de visualización](#)
- [La diferencia entre las funciones input\(\) y raw_input\(\) en Python](#)
- [Listas en Python: operaciones básicas](#)

ARTÍCULOS DE TECNOLOGÍA > DATA SCIENCE

**En Alura encontrarás variados cursos sobre Data Science.
¡Comienza ahora!**

SEMESTRAL

US\$49,90

un solo pago de US\$49,90

- ✓ 218 cursos
- ✓ Videos y actividades 100% en Español
- ✓ Certificado de participación
- ✓ Estudia las 24 horas, los 7 días de la semana
- ✓ Foro y comunidad exclusiva para resolver tus dudas
- ✓ Acceso a todo el contenido de la plataforma por 6 meses

¡QUIERO EMPEZAR A ESTUDIAR!

[Paga en moneda local en los siguientes países](#)

ANUAL

US\$79,90

un solo pago de US\$79,90

- ✓ 218 cursos
- ✓ Videos y actividades 100% en Español
- ✓ Certificado de participación
- ✓ Estudia las 24 horas, los 7 días de la semana
- ✓ Foro y comunidad exclusiva para resolver tus dudas
- ✓ Acceso a todo el contenido de la plataforma por 12 meses

¡QUIERO EMPEZAR A ESTUDIAR!

[Paga en moneda local en los siguientes países](#)

Acceso a todos
los cursos

Estudia las 24 horas,
dónde y cuándo quieras

Nuevos cursos
cada semana

NAVEGACIÓN

PLANES

INSTRUCTORES

BLOG

POLÍTICA DE PRIVACIDAD

TÉRMINOS DE USO

SOBRE NOSOTROS

PREGUNTAS FRECUENTES

¡CONTÁCTANOS!

¡QUIERO ENTRAR EN CONTACTO!

BLOG

PROGRAMACIÓN

FRONT END

DATA SCIENCE

INNOVACIÓN Y GESTIÓN

DEVOPS

AOVS Sistemas de Informática S.A

CNPJ 05.555.382/0001-33

SÍGUENOS EN NUESTRAS REDES SOCIALES



ALIADOS



En Alura somos unas de las Scale-Ups seleccionadas por Endeavor, programa de aceleración de las empresas que más crecen en el país.



Fuimos unas de las 7 startups seleccionadas por Google For Startups en participar del programa Growth Academy en 2021

POWERED BY

CURSOS

Cursos de Programación

Lógica de Programación | Java

Cursos de Front End

HTML y CSS | JavaScript | React

Cursos de Data Science

Data Science | Machine Learning | Excel | Base de Datos | Data Visualization | Estadística

Cursos de DevOps

Docker | Linux

Cursos de Innovación y Gestión

Productividad y Calidad de Vida | Transformación Ágil | Marketing Analytics |
Liderazgo y Gestión de Equipos | Startups y Emprendimiento